

Laboratorio 12

Problema 1:

a) Reducción - β de NOT :

$$\text{True} = \lambda x. \lambda y. x$$

$$\text{False} = \lambda x. \lambda y. y$$

$$\text{NOT} = \lambda b. b \text{ false True}$$

NOT TRUE:

$$(\lambda b. b \text{ false True}) \text{ True}$$

$$\rightarrow \text{True False True}$$

$$\rightarrow (\lambda x. \lambda y. x) \text{ False True}$$

$$\rightarrow \text{False}$$

NOT False:

$$(\lambda b. b \text{ false True}) \text{ False}$$

$$\rightarrow \text{False False True}$$

$$\rightarrow (\lambda x. \lambda y. y) \text{ false True}$$

$$\rightarrow \text{True}$$

b) Recursión y ciclos:

En programación funcional no se usan ciclos, sino recursión.

$$ej: n! = \text{si } n=0 \rightarrow 1, \text{ si no } \rightarrow n * (n-1)! \quad \text{!} \rightarrow \text{factorial}$$

c) cuándo usar programación funcional:

Usar cuando se hacen operaciones matemáticas o transformaciones sobre listas.

No usar cuando se necesita modificar variables o manejar muchos estados.

Ej: usar `map`, `filter`, `reduce` en lugar de ciclos.